

WICC 2014 XVI Workshop de Investigadores en Ciencias de la Computación

Performance de arquitecturas multiprocesador: técnicas de simulación y plataformas reconfigurables.

Diego Encinas^{1,2}, Eduardo Kunysz¹, Alicia Szymanowski¹, Martin Morales¹

¹Instituto de Ingeniería y Agronomía - Universidad Nacional Arturo Jauretche

²Instituto de Investigación en Informática LIDI (III-LIDI) - Facultad de Informática
- UNLP

dencinas@unaj.edu.ar, ejkunysz@gmail.com, aliciaszy@yahoo.com.ar,
martin.morales@unaj.edu.ar

Resumen

El objetivo de esta línea de investigación es el estudio de la performance de las arquitecturas multiprocesador por medio de simulación y plataformas reconfigurables. Enfocando a la obtención de herramientas que permitan predecir la eficiencia del sistema ante posibles escenarios y reconfigurar el sistema en tiempo real. Analizando los diferentes componentes del sistema que pueden influir en las prestaciones significativamente y pueden llegar a modelarse y/o reconfigurarse.

Palabras clave: *Arquitecturas Multiprocesador. Simulación. Sistema E/S paralela. Modelado Orientado al Individuo (MoI). Plataformas reconfigurables para procesamiento paralelo. Lógica programable.*

Contexto

Se presenta una línea de Investigación que es parte del Proyecto de Investigación Científico-Tecnológico “Técnicas de

simulación para modelar arquitecturas multiprocesador” de la Universidad Nacional Arturo Jauretche (UNAJ).

En el tema existe un convenio de colaboración en actividades de Investigación y Postgrado con el Instituto de Investigación en Informática – LIDI y el Centro de Técnicas Analógico-Digitales – CeTAD de la Universidad Nacional de La Plata.

Por otra parte, se tiene financiamiento en el marco del programa “Universidad, Diseño y Desarrollo Productivo” del Ministerio de Educación a través de varios proyectos aprobados en la UNAJ.

Introducción

El crecimiento sostenido en la demanda del poder de cómputo remarca la necesidad de sistemas con enfoques de paralelización masiva y cómputo de alta performance (HPC, High Performance Computing) [1]. Los clusters se han convertido en uno de los enfoques principales para lograr paralelismo a bajo costo. Una noción extendida lo constituye la utilización de grid computing y más recientemente cloud computing. Independientemente de la solución, estos

sistemas constan de un gran número de componentes incluyendo nodos de procesamiento, bancos de memoria, discos, entre otros.

Sistemas de E/S Paralela

El aumento del número de unidades de procesamiento en los clusters, los avances tanto en velocidad como en potencia de las unidades de procesamiento y las crecientes demandas de las aplicaciones científicas que utilizan cómputo de altas prestaciones trae mayores exigencias a los sistemas de E/S paralelas.

En muchos casos, el cuello de botella de los sistemas paralelos es la E/S de estos sistemas dada las exigencias que debe afrontar [2]. La E/S Paralela es esencial para emparejar el avance de las arquitecturas de los procesadores y el rápido crecimiento de la capacidad computacional. Aunque la arquitectura jerárquica de memoria multinivel puede evitar grandes pérdidas de prestaciones debido a los retardos de acceso a disco, la capacidad de memoria es limitada. Además, como la capacidad computacional aumentará, la disponibilidad de memoria por core decrecerá, especialmente si la escala de los sistemas de HPC se proyecta a millones de cores o más. Varias simulaciones científicas y de ingeniería de áreas críticas de investigación, tales como la nanotecnología, astrofísica, clima y energía física están convirtiéndose en aplicaciones intensivas de datos. Para poder disminuir la brecha entre CPUs-E/S se deben identificar los factores que influyen en las prestaciones y proponer nuevas soluciones [3] [4].

En el área de tolerancia a fallas en sistemas de cómputo de alta prestaciones se puede notar la importancia de la unidad de E/S en las arquitecturas paralelas como un punto a mejorar para lograr cubrir las

exigencias de las aplicaciones que utilizan HPC. Una manera de llevar a cabo este trabajo es utilizar técnicas de simulación para evaluar el efecto de los cambios de los factores con mayores influencias en las prestaciones del sistema de E/S paralelo.

Se puede disminuir la complejidad y la probabilidad de errores en la generación de sistemas híbridos desarrollando una simulación específica de éstos utilizando diferentes frameworks [5] [6] [7].

Se propone modelar y simular la arquitectura de E/S paralela, por medio de técnicas de simulación basadas en modelos orientados al individuo (Sistemas Multi-Agente, MAS-Multi Agent Systems), para evaluar el efecto de dimensionar el sistema de E/S o cambiar componentes como la red de almacenamiento, dispositivos de E/S, entre otros.

Plataformas reconfigurables para procesamiento paralelo

Muchos investigadores han demostrado que la computadora de propósitos generales con procesadores gráficos incorporados (GPUs), son una alternativa interesante para resolver problemas de cálculos intensivos. El desafío en estos casos consiste en trasladar el problema concreto a las limitaciones de la programación gráfica. Éstos GPUs (Graphics Processing Unit) , contienen múltiples procesadores simples, en su conjunto llamados many-cores. Este tipo de tecnología normalmente utiliza arquitecturas híbridas, donde parte del procesamiento se realiza en la computadora de propósitos generales y la otra en la placa gráfica.

En los últimos años, se ha buscado expandir este concepto hacia plataformas de procesamiento más específico. Para obtener mayor eficiencia, los fabricantes

de computadoras de altas prestaciones, han introducido unidades de co-procesamiento de arreglos de lógica programable [8]. La aplicación principal se ejecuta en los microprocesadores, mientras que las FPGAs manejan las porciones de código que requieren mayor tiempo de ejecución. Estas porciones de procesamiento suelen ser datos en paralelo superpuestos, arquitecturas que se pueden implementar con una alta granularidad, una sola instrucción, una instrucción y múltiples datos (SIMD), entre otros [9].

Una de las ventajas de este tipo de arquitectura es la capacidad que tiene el procesador principal de reconfigurar las FPGAs en tiempo real. Esto permite la reutilización de partes de hardware en virtud de mejorar la performance del software, lo que crea un nuevo paradigma en el campo de los desarrolladores de aplicaciones. Los sistemas tradicionales utilizan técnicas de descripción de hardware como VHDL. Otros utilizan lenguajes de alto nivel adaptados ([10][11][12][13]) como C o Fortran, o entornos de programación gráficos [14][15]. Esas últimas alternativas permiten una transición natural entre el mundo de hardware y el software.

Con las primeras FPGAs, el reducido tamaño y capacidad que tenían, limitaba mucho la evolución de sistemas tan complejos como los que se pueden implementar en una ASIC (Application-Specific Integrated Circuit). Pero hoy en día sabemos, que los recursos de los dispositivos de arreglos de lógica programable, se incrementan cada año debido a la Ley de Moore, con el agregado de memoria RAM embebida. Con estos incrementos se logran diseños de gran magnitud. Hasta tal punto es así que la tendencia actual es integrar dentro de una FPGA sistemas digitales completos que incluyen un

microprocesador de propósito general y todo el hardware de propósito específico que requiere la aplicación.

Líneas de Investigación, Desarrollo e Innovación

Temas de Estudio e Investigación

- Arquitecturas multiprocesador para procesamiento paralelo: multiprocesador de memoria compartida, multiprocesador on-chip de memoria distribuida. Multicore, Clusters, Clusters de multicore. Grid. Cloud.
- Arquitectura de E/S paralela considerando el software, hardware, comunicaciones entre módulos y dispositivos de almacenamiento.
- Técnicas de comunicación entre dispositivos de lógica programable para optimizar el rendimiento y consumo de las HPRC (High Performance Reconfigurable Computing).

Resultados y Objetivos

Investigación experimental

- Diseño y desarrollo de modelos mediante técnicas de Modelado Orientado al Individuo (MoI) o Sistemas Multi-Agente.
- Utilización de agentes para generar la funcionalidad de los elementos físicos (procesadores, memoria, buses, drivers, entre otros) como así también de las interfaces en las arquitecturas de E/S.
- Generar múltiples dispositivos reconfigurables para implementar

sistemas paralelos con múltiples unidades de procesamiento.

- Simular distintas topologías de interconexión entre unidades de procesamiento reconfigurables para optimizar el camino de la información.

Formación de Recursos Humanos

Dentro de la temática de la línea de I/D se participa en el dictado de la carrera de Ingeniería Informática de la UNAJ. También aportan trabajos de alumnos de las materias Redes de Computadoras 2 y Sistemas Operativos 1 y 2.

En esta línea de I/D existe cooperación a nivel nacional e internacional. Hay 3 investigadores realizando su Doctorado y 1 realizando una Maestría.

Referencias

1. Grama A, Gupta A, Karypis G, Kumar V. "Introduction to parallel computing". Second Edition. Pearson Addison Wesley, 2003.
2. H Hennessy, J. L., Patterson, and D. A., Computer Architecture, Fourth Edition: A Quantitative Approach. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2006.
3. J. M. May, Parallel I/O for high performance computing. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001.
4. V. Balaji, Earth system modelling – Volume 4. IO and Postprocessing. Springer, 2013.
5. D. Encinas, Utilización de un reloj global para el modelado de un ambiente simulado distribuido. XVIII Congreso Argentino de Ciencias de la Computación. 2012
6. D. Encinas, Simulación de una red CAN para dimensionar las comunicaciones de una IMU. VII Congreso Argentino de Tecnología Espacial. 2013.
7. D. Black, SystemC: From the Ground Up. Second Edition, Springer, 2010.
8. V. Kindratenko, A case study in porting a production scientific supercomputing application to a reconfigurable computer. IEEE Symposium on Field-Programmable Computing Machines. 2006.
9. C. Rupp, The NAPA Adaptive Processing Architecture. IEEE Symposium on FPGAs Custom Computing Machines. 1998.
10. "Implementing Simulink Designs on SRC-6 System", David Meixner, Volodymyr Kindratenko, David Pointer, Innovative Systems Laboratory, National Center for Supercomputing Applications, University of Illinois at Urbana-Champaign (<http://www.srccomp.com/carte-programming-environment>)
11. "Using Hardware Libraries with Impulse C", Ralph Bodenner, Director of Product Development, Impulse Accelerated Technologies, Inc. Application Note (<http://www.impulseaccelerated.com/>)
12. "Low Power Hybrid Computing for Efficient Software Acceleration", Mitronics, White Paper. (<http://www.mitronics.com/>)
13. "DK4 - Handel-C Language Reference Manual", Celoxica, (<http://www.celoxica.com/>)
14. "CoreFire™ Design Suite", Annapolis Micro Systems, Inc., DataSheet (<http://www.annapmicro.com/>)
15. "Model-Based Design with Simulink, HDL Coder, and Xilinx System Generator for DSP", Kiran Kintali and Yongfeng Gu, MathWorks, White Paper (<http://www.mathworks.com/fpga-design/simulink-with-xilinx-system-generator-for-dsp.html>)